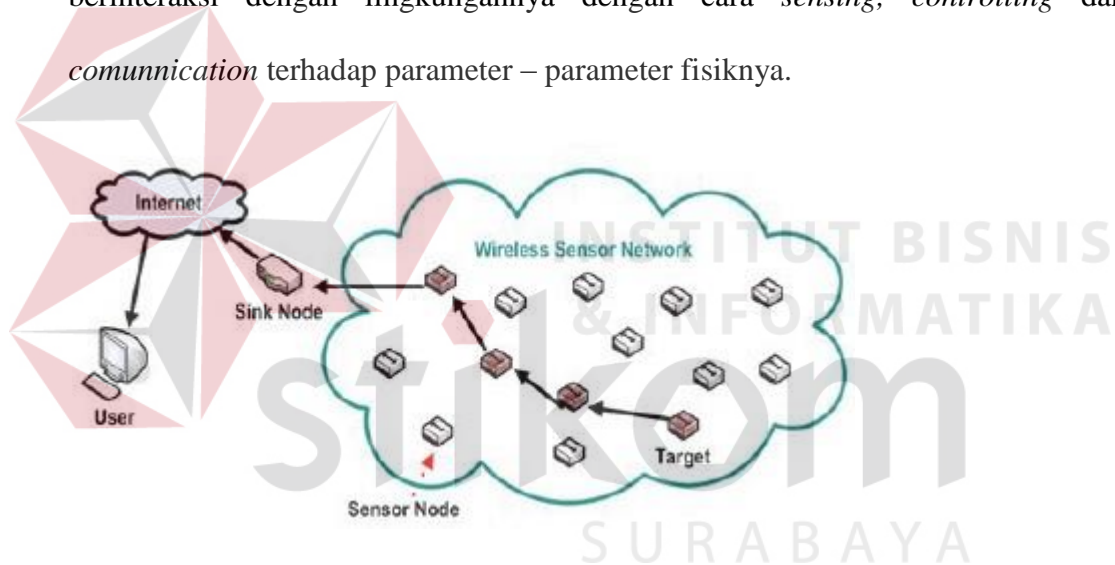


BAB II

LANDASAN TEORI

2.1 *Wireless sensor network*

Wireless sensor network (jaringan sensor nirkabel) terbentuk dari kumpulan titik - titik sensor yang sangat banyak yang bersifat individu dan tersebar tidak beraturan dalam suatu area yang disebut *sensor field*, yang diletakkan di beberapa tempat untuk memonitoring kondisi suatu tempat dan dapat berinteraksi dengan lingkungannya dengan cara *sensing*, *controlling* dan *communication* terhadap parameter – parameter fisiknya.



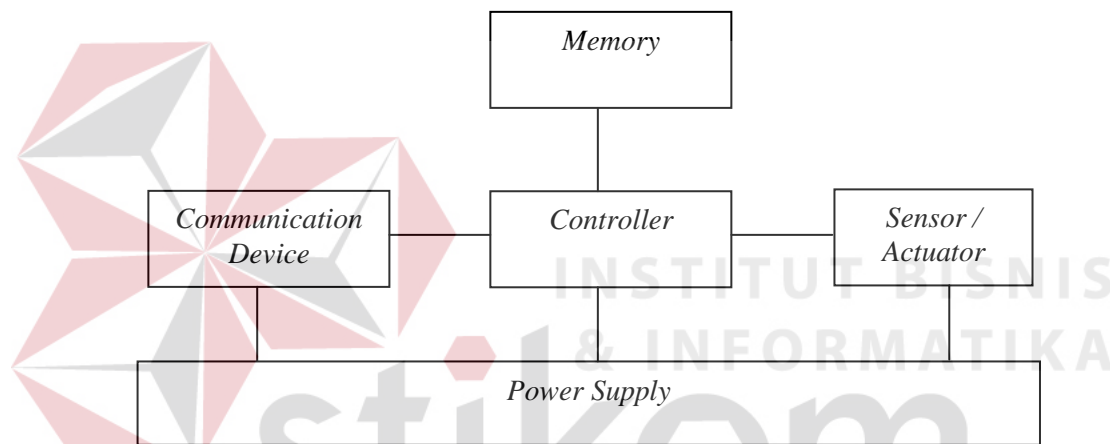
Gambar 2.1 Arsitektur WSN

Sumber : (<http://digilib.tes.telkomuniversity.ac.id>)

Tiap *node* sensor memiliki kemampuan untuk mengumpulkan data dan berkomunikasi dengan *node* sensor lainnya. Peletakan titik-titik *node* sensor tidak perlu direkayasa sedemikian rupa atau ditetapkan sebelumnya (*fixed*). Data yang dikirimkan melalui transmisi radio akan diteruskan menuju BS (*Base Station*) atau

sink node yang merupakan penghubung antara *node* dengan *user*. Informasi tersebut dapat diakses melalui berbagai *platform* seperti koneksi satelit sehingga memungkinkan user untuk mengakses secara *realtime* melalui *remote server*. (Sugiarto, & Sakti, 2009)

Setiap *node* dalam WSN (*Wireless Sensor Network*) terdiri dari lima komponen, yaitu kontroler / mikrokontroler, memori, sensor / aktuator, perangkat komunikasi dan catu daya. Komponen – komponen dari sebuah *node* ditunjukkan pada gambar 2.2 dibawah ini.



Gambar 2.2 Komponen – komponen penyusun *Node* dalam WSN (*Wireless Sensor Network*)

Sumber : (Sugiarto, & Sakti, 2009)

a. *Communication Device*

Berfungsi untuk menerima / mengirim data dengan menggunakan protokol IEEE 802.15.4 atau IEEE 802.11 b/g kepada *device* atau *node* lainnya.

b. *Microcontroller*

Berfungsi untuk melakukan fungsi perhitungan, mengontrol dan memproses *device – device* yang terhubung dengan mikrokontroler.

c. *Sensor*

Berfungsi untuk men-sensing besaran – besaran fisis yang hendak diukur. Sensor adalah suatu alat yang mampu untuk mengubah suatu bentuk energi ke bentuk energi lain, dalam hal ini mengubah energi yang diukur menjadi energi listrik yang kemudian diubah oleh ADC (*Analog to Digital Converter*) menjadi deretan pulsa terkuantasi yang kemudian bisa dibaca oleh mikrokontroler.

d. *Memory*

Berfungsi sebagai bahan tambahan memori bagi sistem *wireless* sensor.

e. *Power Supply*

Berfungsi sebagai sumber energi bagi sistem *Wireless* Sensor secara keseluruhan. (Nugroho, 2014)

2.2 Zigbee

ZigBee adalah spesifikasi untuk jaringan protokol komunikasi tingkat tinggi, menggunakan radio digital berukuran kecil dengan daya rendah, dan berbasis pada standar IEEE (*Institute of Electrical and Electronics Engineers*) 802.15.4-2003 untuk jaringan personal nirkabel tingkat rendah, seperti saklar lampu nirkabel dengan lampu, alat pengukur listrik dengan inovasi *In-Home Display* (IHD), serta perangkat - perangkat elektronik konsumen lainnya yang menggunakan jaringan radio jarak dekat dengan daya transfer data tingkat rendah.

Teknologi yang memenuhi spesifikasi dari ZigBee adalah perangkat dengan pengoperasian yang mudah, sederhana, membutuhkan daya sangat rendah serta biaya yang murah jika dibandingkan dengan WPANs (*Wireless Personal Area Networks*) lainnya, yakni Bluetooth. ZigBee fokus pada aplikasi *Radio Frequency* (RF) yang membutuhkan data tingkat rendah, baterai tahan lama, serta jaringan yang aman (Faludi, 2011).

2.2.1 Topologi Jaringan Zigbee

Zigbee memiliki beberapa *topology* secara umum diantaranya adalah :

a. Pair

Topology pair adalah *topology* yang terdiri dari 2 *node*. Salah satu berupa koordinator dan yang lain berupa router atau *end device*.

b. Star

Topology star adalah *topology* yang memiliki koordinator yang berada di tengah dari *topology* star yang terhubung melingkar dengan *end device*. Setiap data yang lewat selalu melalui koordinator terlebih dahulu. *End device* tidak dapat berkomunikasi secara langsung.

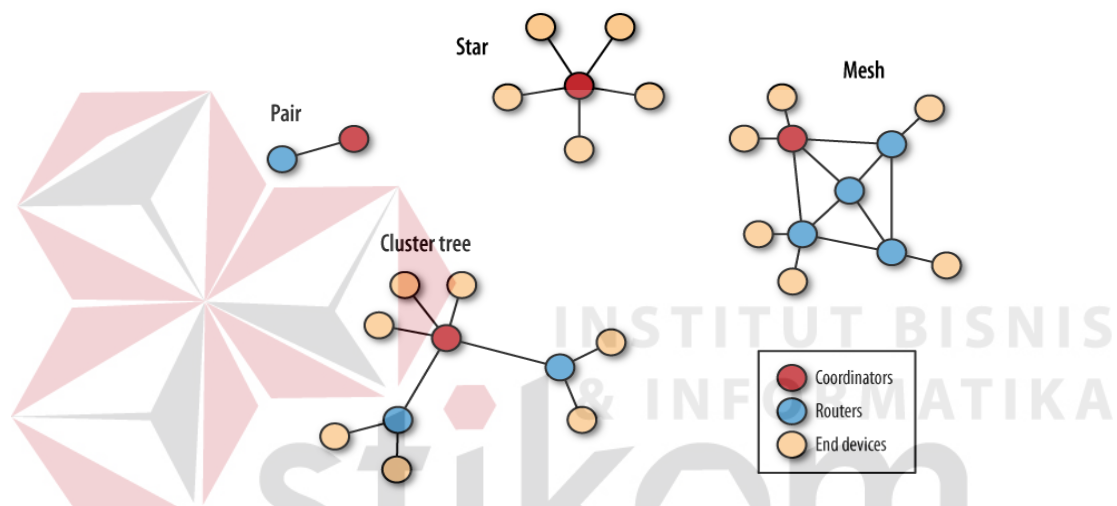
c. Mesh

Topology mesh menugaskan setiap router sebagai koordinator radio. Radio dapat melewati pesan melalui router dan *end device* sesuai dengan kebutuhan. Koordinator bekerja untuk manajemen jaringan. Dapat berupa pesan rute. Berbagai macam *end device* dapat ditambahkan pada koordinator dan router. Dengan ini dapat mengirimkan dan menerima

informasi, tapi masih membutuhkan bantuan “parent’s” untuk dapat berkomunikasi dengan *node* yang lain.

d. *Cluster Tree*

Topology cluster tree menjadikan router sebagai *backbone* dan setiap router terdapat *end device* yang mengililinginya. Konfigurasinya tidak jauh beda dengan *topologi mesh*.



Gambar 2.3 Topology zigbee pair, star, mesh dan cluster tree

Sumber : (Faludi, 2011)

2.3 Arduino

Arduino adalah *prototipe platform* elektroonik *opensource* yang terdiri mikrokontroler, bahasa pemrograman, dan IDE (*Integrated Development Environment*). Arduino adalah alat untuk membuat aplikasi interaktif, yang dirancang untuk mempermudah proyek bagi pemula, tapi masih fleksibel bagi para ahli untuk mengembangkan proyek – proyek yang kompleks. (Banzi, 2009)

2.3.1 Arduino Mega 2560

Arduino mega 2560 adalah papan mikrokontroler berdasarkan ATmega2560 (*datasheet*). Ini memiliki 54 digital pin input / output (pin 15 dapat digunakan sebagai output PWM (*Pulse Width Modulation*)), 16 analog input, 4 UART (*Universal Asynchronous Receiver Transmitter*) (hardware port serial), osilator kristal 16 MHz, koneksi USB, jack listrik, header ICSP (*In-Circuit Serial Programming*), dan tombol reset. Semuanya diperlukan untuk mendukung kerja mikrokontroler, cara mengaktifkan Arduino mega 2560 adalah dengan menghubungkannya ke komputer dengan kabel USB atau memberikan *power* dengan adaptor AC - DC atau baterai. Arduino Mega ini *compatible* dengan Arduino Duemilanove atau Diecimila.

Mega 2560 adalah update dari Arduino Mega. Mega 2560 berbeda dari semua *board* sebelumnya yang tidak menggunakan FTDI (*Future Technology Devices International*) chip driver USB - to -serial. Revisi ke 2 dari *board* Mega 2560 memiliki resistor 8U2, sehingga lebih mudah untuk dimasukkan ke dalam mode DFU (*Device Firmware Update*).

Revisi 3 dari *board* Mega 2560 memiliki fitur-fitur baru berikut :

- 1.0 pinout : menambahkan SDA (*Shouldbe Dhe Ather*) dan pin SCL (*Simply Connect Last*) yang dekat dengan pin AREF dan dua pin baru lainnya ditempatkan dekat dengan pin RESET, IOREF yang memungkinkan *shield* untuk beradaptasi dengan tegangan yang tersedia dari *board*. Nantinya, *shield* akan *compatible* baik dengan *board* yang menggunakan AVR (*Advanced Versatile RISC*), yang beroperasi dengan

5V dan dengan Arduino Due yang beroperasi dengan 3.3V. Yang kedua adalah pin tidak terhubung, yang disediakan untuk tujuan lainnya.

- Sirkuit RESET kuat .
- Atmega 16U2 menggantikan 8U. (Arduino.cc)



Gambar 2.4 Arduino Mega 2560 Sisi Depan (Kiri) dan Belakang(Kanan)

Sumber : (arduino.cc)

Secara umum arduino terdiri dari dua bagian, yaitu:

1. *Hardware*: papan input/output (I/O)
2. *Software*: *software* arduino meliputi IDE untuk menulis program, driver untuk koneksi dengan komputer, contoh program dan *library* untuk pengembangan program. (Djuandi, 2011)

Berikut adalah Tabel 1 spesifikasi dari arduino mega 2560.

Tabel 2.1 Spesifikasi Arduino Mega 2560

| | |
|------------------------------------|-----------------|
| Mikrokontroler | ATMega 2560 |
| Tegangan Operasi | 5V |
| <i>Input</i> tegangan(rekomendasi) | 7 – 12V |
| <i>Input</i> tegangan (Maksimal) | 6 – 20V |
| <i>Digital</i> I/O Pin | 54 (15 pin PWM) |
| <i>Pin input Analog</i> | 16 |

| | |
|---------------------------|--|
| DC current per I/O Pin | 40mA |
| Pin DC Current untuk 3.3V | 50Ma |
| Memori <i>flash</i> | 256Kb, 8Kb digunakan untuk <i>bootloader</i> |
| SRAM | 8Kb |
| EEPROM | 4Kb |
| <i>Clock speed</i> | 16 Mhz |

1. Daya (*Power*)

Arduino mega dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Sumber daya dipilih secara otomatis. Eksternal (non - USB) daya dapat berasal baik dari adaptor AC - DC atau baterai. Adaptor ini dapat dihubungkan dengan menancapkan plug 2.1mm pusat - positif ke colokan listrik *board*. Baterai dapat dimasukkan dalam Gnd dan Vin pin header dari konektor daya.

Board dapat beroperasi pada pasokan eksternal 6 sampai 20 volt. Jika tegangan dengan kurang dari 7V , tegangan pada *board* kemungkinan akan tidak stabil. Jika menggunakan lebih dari 12V , regulator tegangan bisa panas dan merusak *board*. Kisaran yang disarankan adalah 7 sampai 12 volt .

Pin listrik adalah sebagai berikut :

VIN : Tegangan *input* ke *board* Arduino ketika itu menggunakan sumber daya eksternal (ebagai lawan 5 volt dari koneksi USB atau sumber daya diatur lain). Kita dapat memasok tegangan melalui pin ini.

5V : Pin *output* 5V diatur dari regulator di *board*. *Board* dapat diaktifkan dengan daya baik dari colokan listrik DC (7 - 12V) , konektor USB (5V) , atau pin VIN dari board (7-12V). Jika

tegangan diberikan melalui 5V atau 3.3V melewati regulator , dan dapat merusak *board*, maka tidak disarankan.

3V3 : Sebuah pasokan 3,3 volt dihasilkan oleh regulator on-board yang dapat menarik arus maksimum 50 mA.

GND : Pin tanah.

IOREF : Pin pada *board* Arduino memberikan tegangan referensi saat mikrokontroler sedang beroperasi. Sebuah *shield* dikonfigurasi dengan benar agar dapat membaca pin tegangan IOREF dan memilih sumber daya yang tepat atau mengaktifkan penerjemah tegangan pada *output* untuk bekerja dengan 5V atau 3.3V.

(arduino.cc)

2. Memori

ATmega2560 memiliki 256 KB dari *flash memory* untuk menyimpan kode (8 KB digunakan untuk *bootloader*), 8 KB SRAM dan 4 KB EEPROM (yang dapat dibaca dan ditulis dengan *library* EEPROM). (Arduino.cc)

3. Input dan Output

Masing-masing dari 54 digital pin (pin *header*) pada Mega dapat digunakan sebagai *input* atau *output*, dengan menggunakan fungsi dari `pinMode()`, `digitalWrite()`, dan `digitalRead()`. Mereka beroperasi pada tegangan 5V. Setiap pin dapat memberikan atau menerima maksimum 40 mA dan memiliki resistor *pull-up internal* yang (terputus secara *default*) dari 20-50 KOhms. Selain itu, beberapa pin memiliki fungsi khusus :

Serial : 0(RX) dan 1(TX); Serial 1 : 19(RX) dan 18(TX) ; Serial 2 : 17(RX) dan 16 (TX) ; Serial 3 : 15 (RX) dan 14(TX). Yang digunakan untuk menerima (RX) dan mengirimkan (TX) TTL data serial. Pin 0 dan 1 juga terhubung ke pin yang sesuai dari ATmega16U2 USB - to- TTL *chip* Serial. Letak PIN serial dapat dilihat pada gambar 2.25.

Eksternal Interupsi: 2 (*interrupt* 0) , 3 (*interrupt* 1) , 18 (*interrupt* 5) , 19 (*interrupt* 4) , 20 (*interrupt* 3) , dan 21 (*interrupt* 2). Pin ini dapat dikonfigurasi untuk memicu *interrupt* pada nilai yang rendah, naik atau jatuh tepi, atau perubahan nilai. Lihat *AttachInterrupt()* fungsi untuk rincian. Letak PIN interupsi dapat dilihat pada gambar 2.25.

PWM: Pin 2-13 dan 44 sampai 46. Menyediakan 8 - bit PWM *output* dengan *analogWrite()* *function*. Letak PIN PWM dapat dilihat pada gambar 2.25.

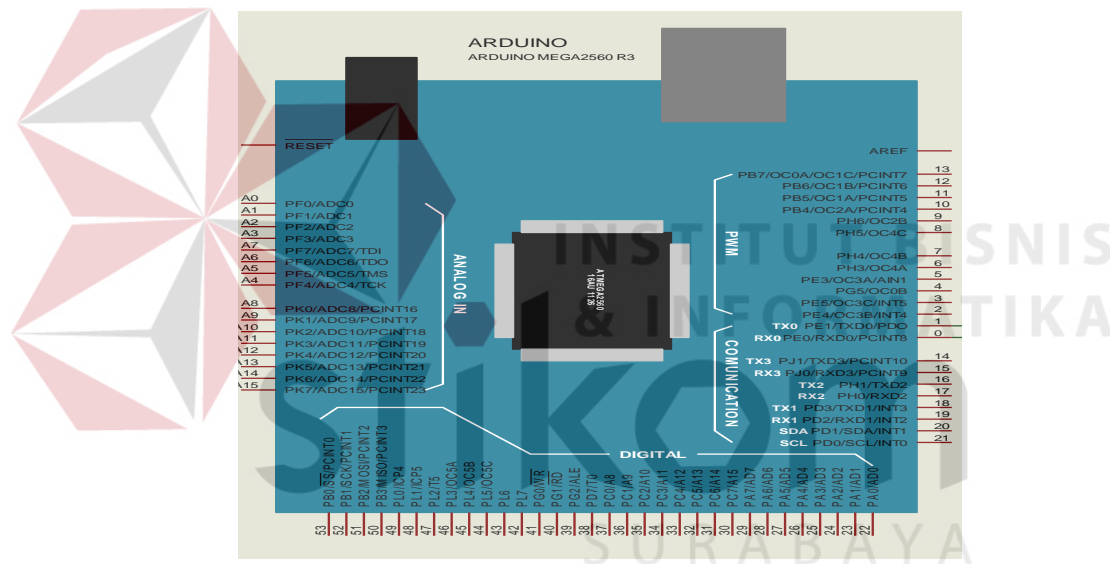
SPI (Serial Peripheral Intervace) : 50(MISO), 51(MOSI), 52(SCK), 53(SS). Pin ini mendukung komunikasi SPI menggunakan *library* SPI. Pin SPI juga pecah pada header ICSP, yang secara fisik kompatibel dengan Uno , Duemilanove dan Diecimila. Letak PIN SPI dapat dilihat pada gambar 2.25.

LED : 13. Ada *built -in* LED terhubung ke pin digital 13. Ketika pin dengan nilai TINGGI , LED menyala , ketika pin yang dipakai RENDAH , lampu akan mati.

TWI (Two – Ware Inteerface): Pin 20(SDA) dan pin 21(SCL). Dukungan komunikasi TWI menggunakan *library* Wire. Yang perly diperhatikan

adalah bahwa pin ini tidak berada di lokasi yang sama dengan pin TWI pada Duemilanove atau Diecimila. Letak PIN TWI dapat dilihat pada gambar 2.25.

Arduino Mega2560 memiliki 16 input analog , yang masing-masing menyediakan 10 bit resolusi (yaitu 1024 nilai yang berbeda). Secara *default* mereka mengukur dari *ground* sampai 5 volt , meskipun mungkin untuk mengubah jangkauan menggunakan pin AREF dan fungsi `analogReference()` .
(Arduino.cc)

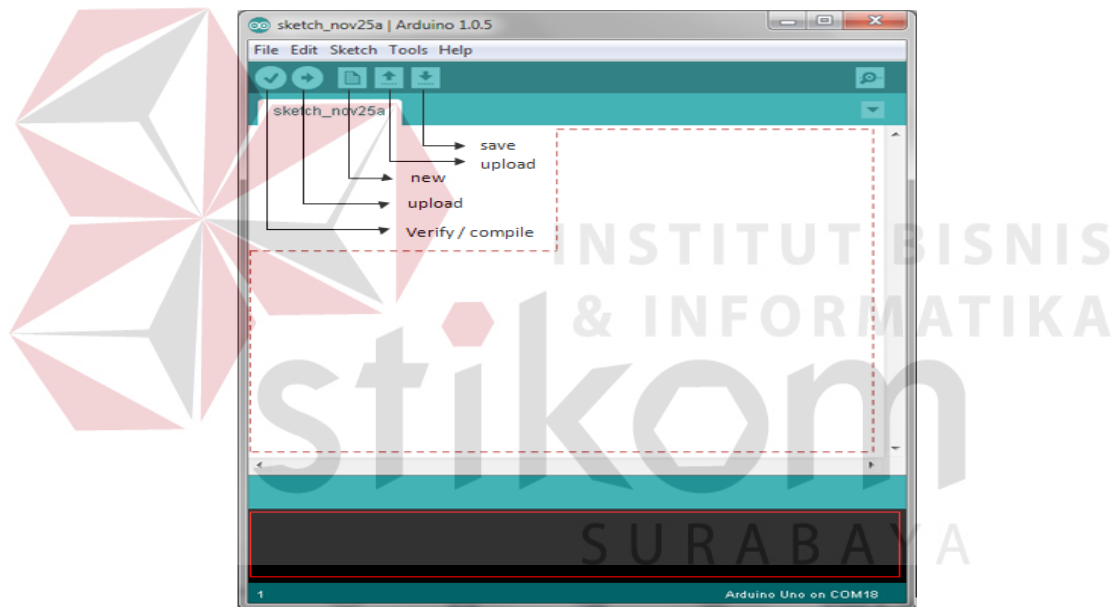


Gambar 2.5 Interface Arduino 2650

2.4 Software Arduino IDE

Arduino IDE adalah *software* yang ditulis menggunakan java dan berdasarkan pengolahan seperti, avr-gcc, dan perangkat lunak *open source* lainnya (Djuandi, 2011). Arduino IDE terdiri dari:

1. Editor program, sebuah *window* yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *processing*.
2. *Verify / Compiler*, sebuah modul yang mengubah kode program (bahasa *processing*) menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa *processing*, yang dipahami oleh mikrokontroler adalah kode biner.
3. *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam memori mikrokontroler di dalam papan arduino.



Gambar 2.6 Tampilan *Software* Arduino IDE

(Arduino, 2011)

Pada Gambar 2.6 terdapat *menu bar*, kemudian *toolbar* dibawahnya, dan sebuah area putih untuk *editing sketch*, area hitam dapat kita sebut sebagai *progress area*, dan paling bawah dapat kita sebut sebagai “*status bar*”.

2.5 Bahasa Pemograman Arduino

Arduino ini bisa dijalankan di komputer dengan berbagai macam *platform* karena didukung atau berbasis Java. *Source* program yang dibuat untuk aplikasi mikrokontroler adalah bahasa C/C++ dan dapat digabungkan dengan *assembly*. (Arduino.cc)

1. Struktur

Setiap program Arduino (biasa disebut *sketch*) mempunyai dua buah fungsi yang harus ada (Arduino.cc). Antara lain:

a) `void setup() { }`

Semua kode didalam kurung kurawal akan dijalankan hanya satu kali ketika program Arduino dijalankan untuk pertama kalinya.

b) `void loop() { }`

Fungsi ini akan dijalankan setelah setup (fungsi *void setup*) selesai. Setelah dijalankan satu kali fungsi ini akan dijalankan lagi, dan lagi secara terus menerus sampai catu daya (*power*) dilepaskan.

2. Serial

Serial digunakan untuk komunikasi antara arduino *board*, komputer atau perangkat lainnya. Arduino *board* memiliki minimal satu *port* serial yang berkomunikasi melalui pin 0 (RX) dan 1 (TX) serta dengan komputer melalui USB. Jika menggunakan fungsi – fungsi ini, pin 0 dan 1 tidak dapat digunakan untuk *input* digital atau *output* digital (Arduino.cc). Terdapat beberapa fungsi serial pada arduino, antara lain:

a. *Syntax*

Adalah elemen bahasa C yang dibutuhkan untuk format penulisan.

(Arduino.cc)

b. Variabel

Sebuah program secara garis besar dapat didefinisikan sebagai instruksi untuk memindahkan angka dengan cara yang cerdas. Variabel inilah yang digunakan untuk memindahkannya. (Arduino.cc).

2.6 Xbee

Xbee merupakan perangkat yang menunjang komunikasi data tanpa kabel (*wireless*). Terdapat 2 jenis Xbee, yaitu :

a. Xbee 802.15.4 (*Xbee Series 1*)

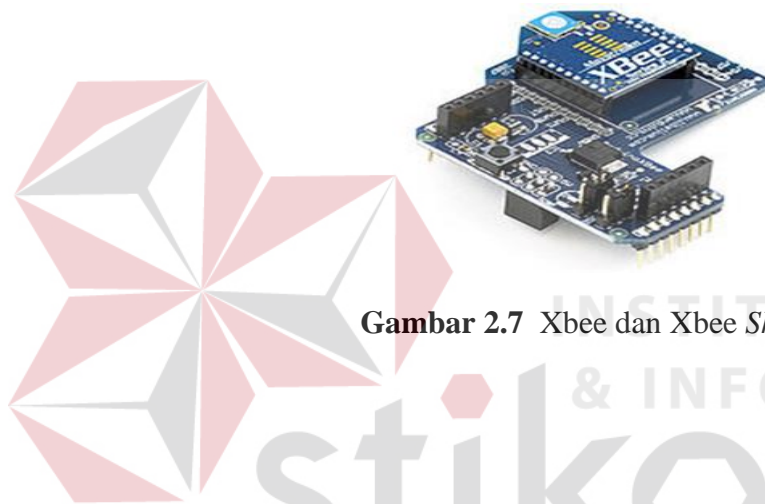
Xbee seri ini hanya dapat digunakan untuk komunikasi *point to point* dan topologi *star* dengan jangkauan 30 meter *indoor* dan 100 meter *outdoor*.

b. Xbee ZB *Series 2*

Xbee *series 2* dapat digunakan untuk komunikasi *point to point*, *point to multipoint* dan topologi *star*, dan topologi *mesh* dengan jangkauan 40 meter *indoor* dan 100 meter *outdoor*.

Xbee *series 1* maupun *series 2* tersedia dalam 2 bentuk berdasarkan kekuatan transmisinya yaitu xbee *reguler* dan xbee-pro. Xbee-PRO mempunyai kekuatan transmisi lebih kuat, ukuran perangkatnya lebih besar, dan harganya lebih mahal. Xbee-PRO mempunyai jangkauan *indoor mencapai* 60 meter dan

outdoor mencapai 1500 meter. Xbee ini dapat digunakan sebagai pengganti serial / USB atau dapat memasukkannya ke dalam *command mode* dan mengkonfigurasinya untuk berbagai macam jaringan *broadcast* dan mesh. Shield membagi setiap pin Xbee. Xbee juga menyediakan header pin *female* untuk penggunaan pin digital 2 sampai 7 dan input analog, yang di *cover* oleh shield (pin digital 8 sampai 13 tidak tercover oleh shield, sehingga dapat menggunakan header pada papan itu sendiri. (Arduino, 2011)



Gambar 2.7 Xbee dan Xbee Shield.

(Arduino, 2011)

Berikut parameter untuk mengkonfigurasi modul Xbee S2 dengan mode AT.

Tabel 2.2 Parameter Xbee

| Perintah | Keterangan | Nilai <i>valid</i> | Nilai <i>Default</i> |
|-----------|--|------------------------------|----------------------------|
| ID | <i>Id</i> jaringan modul Xbee | 0-0Xffff | 3332 |
| CH | Saluran dari modul Xbee. | 0x0B-0x1A | 0x0C |
| SH dan SL | Nomor seri modul Xbee(SH memberikan 32bit tinggi, SL32 bit rendah). <i>Read-only</i> . | 0-0xFFFFFFFF | berbeda untuk setiap modul |
| MY | Alamat 16-bit dari modul. | 0-0xFFFF | 0 |
| Dh dan DL | Alamat tujuan untuk komunikasi nirkabel(DH adalah 32bit tinggi, DL32 low). | 0-0xFFFFFFFF (untuk kedua | 0(untuk kedua DH dan DL) |

| Perintah | Keterangan | Nilai <i>valid</i> | Nilai <i>Default</i> |
|----------|---|---|----------------------|
| | | DH dan DL) | |
| BD | <i>baud rate</i> yang digunakan untuk komunikasi serial dengan papan Arduino atau komputer. | 0 (1200 bps) 1 (2400 bps) 2 (4800 bps) 3 (9600 bps) 4 (19200 bps) 5 (38400 bps) 6 (57600 bps) 7 (115200 bps) | 3 (9600 baud) |

Catatan : meskipun nilai-nilai yang *valid* dan standar dalam Tabel di atas ditulis dengan awalan "0x" (untuk menunjukkan bahwa mereka adalah nomor heksadesimal), modul tidak akan mencakup "0x" ketika melaporkan nilai parameter, dan anda harus menghilangkannya ketika menetapkan nilai-nilai (Arduino, 2011).

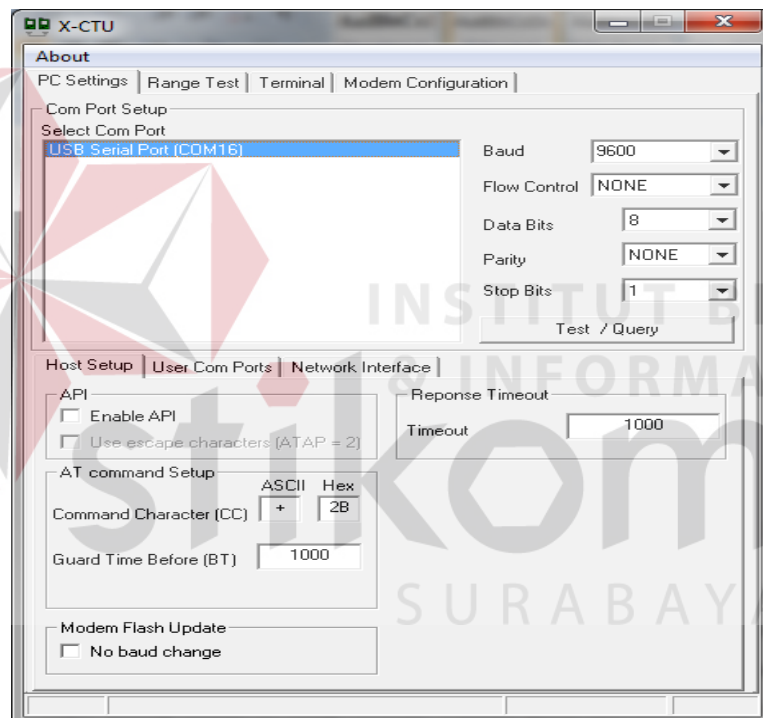
2.7 Xbee Usb Adapter dan Software X-CTU

Xbee USB adapter (Gambar 2.8) merupakan alat untuk menghubungkan modul Xbee ke komputer dengan kabel mini USB dan selanjutnya dapat dikonfigurasi menggunakan *software* X-CTU (Gambar 2.9). *Software* X-CTU merupakan *software* yang digunakan untuk mengkonfigurasi Xbee agar dapat berkomunikasi dengan Xbee lainnya. Parameter yang harus diatur adalah PAN ID (*Personal Area Network*) ID yaitu parameter yang mengatur radio mana saja yang dapat berkomunikasi, agar dapat berkomunikasi PAN ID dalam satu jaringan harus sama. Xbee dapat berkomunikasi *point to point* dan *point to multipoint* (*broadcast*). (Arduino, 2011)



Gambar 2.8 Xbee Usb Adapter dan Kabel Mini Usb

(Arduino, 2011)



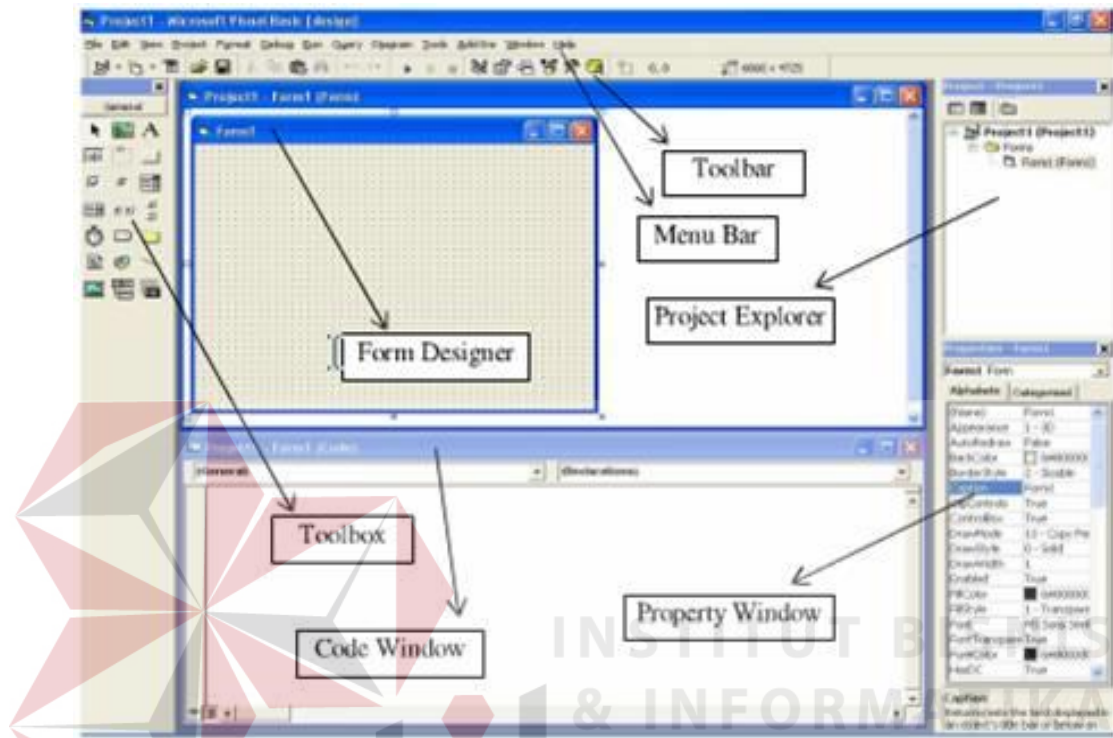
Gambar 2.9 Tampilan Software X-CTU

(Arduino, 2011)

2.8 Visual Basic

Visual Basic adalah salah satu *development tools* untuk membangun aplikasi dalam lingkungan *Windows*. Dalam pengembangan aplikasi, *Visual Basic*

menggunakan pendekatan *Visual* untuk merancang *user interface* dalam bentuk *form*. Tampilan *Visual Basic* terdapat pada *Integrated Development Environment* (IDE) seperti pada Gambar 2.10.



Gambar 2.10 Tampilan Utama Visual Basic 6.0

Sumber : (Octovhiana, 2003)

Adapun penjelasan jendela-jendela adalah sebagai berikut :

- a) *Menu Bar*, digunakan untuk memilih tugas-tugas tertentu seperti menyimpan project, membuka project, dll
- b) *Main Toolbar*, digunakan untuk melakukan tugas-tugas tertentu dengan cepat.
- c) Jendela *Project*, jendela berisi gambaran dari semua modul yang terdapat dalam aplikasi.

- d) Jendela *Form Designer*, jendela merupakan tempat anda untuk merancang user interface dari aplikasi.
- e) Jendela *Toolbox*, jendela berisi komponen-komponen yang dapat anda gunakan untuk mengembangkan user interface.
- f) Jendela *Code*, merupakan tempat bagi anda untuk menulis koding. Anda dapat menampilkan jendela dengan menggunakan kombinasi Shift-F7.
- g) Jendela *Properties*, merupakan daftar properti-properti object yang sedang terpilih. Sebagai contohnya anda dapat mengubah warna tulisan (*foreground*) dan warna latar belakang (*background*). Anda dapat menggunakan F4 untuk menampilkan jendela properti.
- h) Jendela *Color Palette*, adalah fasilitas cepat untuk mengubah warna suatu object.
- i) Jendela *Form Layout*, akan menunjukkan bagaimana form bersangkutan ditampilkan ketika runtime. (Octovhiana, 2003)

2.9 Parameter QoS (*Quality of Service*)

Performansi mengacu ke tingkat kecepatan dan keandalan penyampaian berbagai jenis beban data di dalam suatu komunikasi. Performansi merupakan kumpulan dari beberapa parameter besaran teknis, yaitu :

- *Throughput*, yaitu kecepatan (*rate*) transfer data efektif, yang diukur dalam bps. *Troughput* merupakan jumlah total kedatangan paket yang sukses yang diamati pada *destination* selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut.

$$\text{Throughput} = \frac{\text{Paket data diterima}}{\text{Lama pengamatan}}$$

- *Packet Loss*, merupakan suatu parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah total paket yang hilang, dapat terjadi karena *collision* dan *congestion* pada jaringan dan hal ini berpengaruh pada semua aplikasi karena *retransmisi* akan mengurangi efisiensi jaringan secara keseluruhan meskipun jumlah *bandwidth* cukup tersedia untuk aplikasi-aplikasi tersebut. Umumnya perangkat jaringan memiliki *buffer* untuk menampung data yang diterima. Jika terjadi kongesti yang cukup lama, *buffer* akan penuh, dan data baru tidak akan diterima.

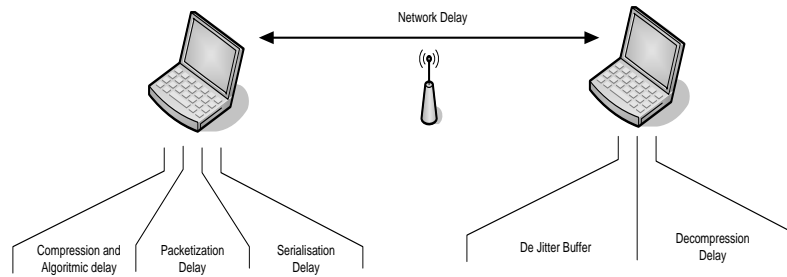
Tabel 2.3 *Packet Loss*

| KATEGORI DEGRADASI | PACKET LOSS |
|--------------------|-------------|
| Sangat bagus | 0 |
| Bagus | 3 % |
| Sedang | 15 % |
| Jelek | 25 % |

- *Delay (latency)*, adalah waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. *Delay* dapat dipengaruhi oleh jarak, media fisik, kongesti atau juga waktu proses yang lama. (Nurhayati, 2012)

$$\text{Delay rata - rata} = \frac{\text{Total delay}}{\text{Total paket yang diterima}}$$

Adapun komponen *delay* adalah sebagai berikut:



Gambar 2.11 *Network Delay*

Tabel 2.4 *Komponen Delay*

| Jenis Delay | Keterangan |
|---------------------------------|---|
| <i>Algorithmic delay</i> | <i>Delay</i> ini disebabkan oleh standar <i>codec</i> yang digunakan. Contohnya, <i>Algorithmic delay</i> untuk G.711 adalah 0 ms |
| <i>Packetization delay</i> | <i>Delay</i> yang disebabkan oleh peng-akumulasian bit <i>voice sample</i> ke <i>frame</i> . Seperti contohnya, standar G.711 untuk <i>payload</i> 160 bytes memakan waktu 20 ms. |
| <i>Serialization delay</i> | <i>Delay</i> ini terjadi karena adanya waktu yang dibutuhkan untuk pentransmisiian paket IP dari sisi <i>originating</i> (pengirim). |
| <i>Propagation delay</i> | <i>Delay</i> ini terjadi karena perambatan atau perjalanan. Paket IP di media transmisi ke alamat tujuan. Seperti contohnya <i>delay</i> propagasi di dalam kabel akan memakan waktu 4 sampai 6 μ s per kilometernya. |
| <i>Coder (Processing) Delay</i> | Waktu yang diperlukan oleh Digital Signal Processing (DSP) untuk mengkompres sebuah block PCM (<i>Pulse – Codec Modulation</i>), nilainya bervariasi bergantung dari <i>codec</i> dan kecepatan prosesor |